German Question Tags: A Computational Analysis

Yulia Clausen

Germanistisches Institut, Ruhr-Universität Bochum, Germany

Abstract

The German language exhibits a range of question tags that can typically, but not always, be substituted for one another. Moreover, the same words can have other meanings while occurring in the sentence-final position. The tags' felicity conditions were addressed in previous corpus-based and experimental work and attributed to semantic and pragmatic properties of tag questions. This paper addresses the question of whether and to what extent the differences among German tags can be determined automatically. We assess the performance of three pretrained German BERT models on a tag question dataset and fine-tune one of these models on the tag word prediction task. A close examination of this model's output indicates that BERT can identify properties relevant for the tags' felicity conditions and interchangeability consistent with previous studies.

Keywords

tag questions, German, tags, annotation, BERT, clustering

1. Introduction

This study provides a computational analysis of German question tags. A (question) tag is a fixed expression that attaches to an utterance (anchor) and is used to elicit a confirmational response from the addressee regarding the anchor proposition. The whole construction is referred to as a tag question (TQ). TQs are a widely studied phenomenon, however, a comprehensive analysis of German tags has been proposed only in recent studies [5, 3, 4]. The German language offers a large variety of invariable tags that can be used interchangeably in some contexts (1), but not in others (2).[1]

(1) Lina says to her sister as they go out of the cinema:
Der Film war gut, ne?/nicht?/oder?
'The film was good, wasn't it?'

(2) Lina comes back from the movies and says to her sister (who did not want to come):
*Der Film war gut, ne!/nicht!/*oder!*
'The film was good, you know!'

CHR 2023: Computational Humanities Research Conference, December 6 – 8, 2023, Paris, France
yulia.clausen@rub.de (Y. Clausen)
0009-0009-0726-0255 (Y. Clausen)

[1] We focus on the use of tags within Germany and do not consider the regional/country-specific differences or individual preferences in tag usage. We work with the existing corpus data to study the general interchangeability potential of tags. In view of this, we assume that if different individuals use different tags in the same sentence to convey the same meaning, these tags are interchangeable in this particular context.

In (1), different tags are equally suitable for requesting confirmation of whether Lina's sister also liked the film. In (2), however, Lina's sister is requested to confirm her acknowledgment of the provided information, in which case *oder* is infelicitous [cf. 4].

Felicity conditions of German tags were addressed in previous experimental and corpus-based studies, and several factors were identified as crucial for the tags' (non-)interchangeability. Among those are syntactic and semantic properties of TQs, as well as pragmatic inferences arising from various contextual aspects (see Section 2 for details). In this study, we pursue the question of whether the similarities and/or differences among tags, and hence cases of their potential interchangeability, can be modeled automatically. Language models, such as BERT [6], are known for their capacity to leverage semantic and other types of linguistic information from the context around a given word (see e.g., [17] for an overview). Therefore, we test whether and how well BERT can identify the properties of German tags, such as those defined in previous work, and whether we can gain new insights from this into the tags' felicity conditions.

It is worth noting that there exists another TQ-relevant distinction in German: Words functioning as tags can have other meanings while occurring in the tag position (i.e., end of a sentence). For example, *nicht* is also a negation particle (e.g., *Kennst du das* ***nicht****?* 'Don't you know that?'). This is a different kind of distinction, since semantically TQs differ considerably from other sentence types ending with the same word. We thus include both types of sentences in our analysis. We expect the sentence type distinction to be easier for BERT than determining the differences among individual tags. The latter, however, is of primary interest to us.

Our paper makes the following contributions. We test the capacities of three existing pretrained German BERT models to differentiate among question tags as well as between TQs and other sentence types. We find that while most models capture the sentence type distinction quite well, they struggle with semantic/pragmatic differences within the tag class. Instead, BERT demonstrates a strong dependence on structural features, such as punctuation. We apply K-Means clustering to the embeddings produced by one of these models and test the overlap of the generated clusters with the linguistic properties of TQs defined in previous work. We find indications as to which of those properties are relevant for the tags' felicity conditions in accordance with previous findings. Finally, we fine-tune the selected model on the next word prediction task with respect to two aspects: prediction of the word class (tag vs. no-tag) and form (e.g., *oder* vs. *ne*). Our experiments show that the fine-tuned model outperforms the original one in both tasks, while at the same time revealing the importance of the dataset size for meaningful prediction.

2. Related work

2.1. German question tags

The meaning and felicity conditions of German tags were addressed in recent corpus-based and experimental studies [5, 3, 4, 9]. Several semantic/pragmatic as well as syntactic factors were found crucial for the tags' felicity conditions and their interchangeability potential. Anchor clause type and speech act provide certain indications regarding the tags' felicity, such that, e.g., imperative directives as in (3) are compatible only with *ja* [cf. 4].

(3) Max wants to play football with his friends, but his father says:
*Mach erst deine Hausaufgaben, ja!/*ne!/*nicht!/*oder!*
'Do your homework first!'[2]

Oftentimes, additional context is required, though. For example, the TQ anchors in (1) and (2) in Section 1 are both declarative assertions, but *oder* is felicitous only in the former. In such cases, information about the interlocutors' epistemic authority provides additional clues, e.g., whether the speaker is informing the addressee or asking for a confirmation (cf. statements vs. questions as functions of TQs in [12]). If the speaker is the source of information, the use of *oder* is typically ruled out. Further constraints are provided by the type of requested confirmation, i.e., the aspect of the anchor proposition the addressee is requested to confirm (*target of confirmation* in [4, 20]). An example would be agreement with the speaker's opinion vs. acknowledgment of the provided information in (1) vs. (2) in Section 1.

These linguistic properties have been found to correlate with different tags as well as with each other to varying degrees ([4], p. 26), and while some of them are straightforward (e.g., anchor clause type), other are more complex and need to be inferred from the context (e.g., target of confirmation).

2.2. Language modeling

Among the growing amount of work on the next word prediction with language models, several studies have focused on linguistic elements in the sentence-final position. Kato, Miyata, and Sato [11] use BERT to generate simplified substitutions for Japanese sentence-ending predicates. Li, Grissom II, and Boyd-Graber [13] predict sentence-final verbs for German and Japanese with neural models for two tasks: predicting the exact verb and a semantically similar one. Mandokoro, Oka, Matsushima, Fukada, Yoshimura, Kawahara, and Tanaka [15] train a BERT model on the task of Japanese sentence-final particle prediction.

Ettinger [7] explores the role of different types of information in prediction of the sentence-final word on the basis of its left-side context for English. Similarly, we implement the tag word prediction task informed only by its left-side context. The factors tested in [7] are similar to those that play a role in the felicity conditions of German tags: semantic roles, event knowledge, and pragmatic inferences. Ettinger finds them to be particularly challenging for BERT.

To our knowledge, there are no studies that explore the features of question tags or focus on automatic tag prediction with language models.

3. Data

We work with the TQ dataset from [4] built from three German corpora: CallHome (CH) [10], OpenSubtitles (OS) [14], and Twitter (TW) [19]. This dataset contains automatically extracted TQ candidates that need to be manually disambiguated as to whether or not they end with a tag. We confine our analysis to the four most frequent tags (*ja*, *ne*, *nicht* and *oder*), for which we unified and annotated the data with the tag/no-tag labels. The annotation was performed

[2]We find that the sense of non-negotiability conveyed by this utterance is best expressed without a tag in English.

by four annotators: the author of this paper and three annotators with a linguistic background. The latter were provided with the annotation guidelines. To ensure the annotation quality, the author of this paper independently annotated approx. 1,000 TQ candidates from each annotator's file. High inter-annotator agreement was reached on these data subsets: Cohen's kappa score of 0.9 with annotator 1 and 0.78 with annotator 2.[3] Any conflicting annotations in these data subsets, i.e., between the author of the paper and each respective annotator, were resolved afterwords. Table 1 shows the number of annotated tag words per corpus used in this study.[4]

Table 1
Distribution of the tag and no-tag instances per tag and corpus.

	CH		OS		TW		Total	
	tag	no-tag	tag	no-tag	tag	no-tag	tag	no-tag
ja	139	283	11,660	280	492	1,034	12,291	1,597
ne	665	2	373	1	640	44	1,678	47
nicht	223	278	4,852	17,719	293	13,077	5,368	31,074
oder	90	0	1,087	0	1,396	0	2,573	0
Total	1,117	563	17,972	18,000	2,821	14,155	21,980	32,718

4. Tag word embeddings

We test the following existing pretrained German BERT models:

1. *bert-base-german-cased*[5] trained on a German Wikipedia dump, Open Legal Data dump, and news articles (12 GB)
2. *bert-base-german-dbmdz-cased*[6] trained on a Wikipedia dump, EU Bookshop corpus, Open Subtitles, CommonCrawl, ParaCrawl, and News Crawl (16 GB)
3. *gbert-large*[7] trained on the OSCAR corpus, Wikipedia dump, the OPUS project, and Open Legal Data (163.4 GB)

To generate the tag word embeddings, we extracted one TQ candidate from each record in the dataset.[8] Depending on the corpus, we applied different preprocessing steps to the extracted TQ candidates. For CH and OS, we removed all meta-language sequences. For TW, we stripped URLs (end of sentence), hashtags and @USERNAME mentions (beginning and end of sentence), and common emoticons (anywhere in sentence). Furthermore, we excluded TQ candidates consisting of fewer than three tokens including the tag word, in order to eliminate (most of)

[3]We could not calculate the inter-annotator agreement with annotator 3, as they did not complete their annotations, so that there were no overlapping annotations available for comparison. The annotation in this case was completed by the author of the paper.
[4]The annotated dataset and the annotation guidelines are available via the Open Science Framework: https://osf.io/pcng9.
[5]https://huggingface.co/bert-base-german-cased
[6]https://github.com/dbmdz/berts#german-bert
[7]https://huggingface.co/deepset/gbert-large; [2].
[8]Some records consist of several sentences (e.g., a tweet) and hence can contain more than one TQ candidate. We extracted each record's first sentence ending with one of the relevant tag words.

the short sequences bearing little meaning. Finally, we removed all duplicates based on case-sensitive string comparison. Examples of the preprocessed sentences in the final dataset are given in Table 2.

Table 2
Examples of TQ candidates before and after preprocessing. Tag words are marked in **bold**.

CH	tag	before	{laugh} Aber das geht erst, das geht nicht ab sechzig, **ja**? Das {laugh}
		after	Aber das geht erst, das geht nicht ab sechzig, **ja**?
	no-tag	before	{laugh} (( )) wahrscheinlich noch besser **ja**.
		after	wahrscheinlich noch besser **ja**.
OS	tag	before	[Deacon] Ein schöner Bau, **nicht**?
		after	Ein schöner Bau, **nicht**?
	no-tag	before	- (Walter) Mr. Taransky will das **nicht**.
		after	Mr. Taransky will das **nicht**.
TW	tag	before	@USERNAME Yo...warum Schwer wenn's auch Einfach geht **ne**.... :P
		after	Yo...warum Schwer wenn's auch Einfach geht **ne**....
	no-tag	before	Ich wollt vor #btn noch eine Rauchen, aber **ne**.
		after	Ich wollt vor #btn noch eine Rauchen, aber **ne**.

We fed the preprocessed TQ candidates through each model and obtained embeddings consisting of either 12 layers with 768 dimensions (*bert-base-german-cased* and *bert-base-german-dbmdz-cased*) or 24 layers with 1,027 dimensions (*gbert-large*) per token. To get a single embedding per token, we concatenated each token's last four layers, thus obtaining one vector with 3,072 (*bert-base-german-cased* and *bert-base-german-dbmdz-cased*) or 4,096 (*gbert-large*) dimensions. Finally, we extracted each tag word's embedding, which we use here as its contextual representation.

5. BERT model comparison

This section discusses the output of the three BERT models with respect to the tag/no-tag distinction and the differences among the tag forms. We reduce the embeddings to three components with Principal Component Analysis (PCA)[9] and map them into a vector space. We use the visualized data for our analysis and provide a more compact version of the plots in Appendix A for illustration.[10]

5.1. *bert-base-german-cased*

This model differentiates prima facie well among the four tag words: Vectors representing the same tags are densely grouped together, while distinct tags are visibly separated from each other (Figure 1a, 2a, 3a in Appendix A). However, each vector group is a tag/no-tag mixture

[9]We used the *scikit-learn* implementation: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html.

[10]The plots in Appendix A were created with *seaborn* (https://seaborn.pydata.org/). The interactive 3D plots used for our analysis were created with *matplotlib* (https://matplotlib.org/) and are available via the Open Science Framework: https://osf.io/pcng9.

(except for *oder*, which has no no-tag counterparts). This suggests that this model only differentiates between the surface forms of the tag words, and will most likely be insufficient in handling finer-grained distinctions, such as different types of utterances ending with the same word.

5.2. *bert-base-german-dbmdz-cased*

The vector groups generated by this model are less dense and have visually less space between them compared to *bert-base-german-cased* (Figure 1b, 2b, 3b in Appendix A). Nonetheless, the model differentiates well among the tags and provides a reasonable tag/no-tag separation in most cases. Furthermore, it subdivides the tag groups, which is not the case with *bert-base-german-cased*. This is particularly prominent for CH (*ja*, *ne* and *nicht*) and TW (all tags).

We find that the formation of subgroups (among the TQs ending with the same tag) is tied to punctuation. Tags are placed into different subgroups depending on whether they are followed by a question mark or a period. This is consistent across the tags and corpora. The tag-preceding comma also plays a role: The tags are either clearly separated (e.g., ‘, ja?’ vs. ‘ja?’ in OS/TW), or there is a gradual transition from one punctuation type to another within a subgroup (e.g., ‘ne.’ vs. ‘, ne.’ in CH).

The tag/no-tag groups typically partially overlap in cases of matching punctuation (e.g., *ja* in OS). Given that tags with different punctuation form distinct subgroups, this suggests that the model considers tags and no-tags with the same punctuation to be more similar than the same tags with different punctuation. Thus, structural features seem to dominate over potential syntactic/semantic differences between TQs and other sentence types ending with the same tag word.

5.3. *gbert-large*

This model falls in between the other two, as its output looks similar to that of *bert-base-german-cased* in terms of compact, spatially well-separated vector groups, while at the same time providing a good tag/no-tag distinction akin to *bert-base-german-dbmdz-cased* (Figure 1c, 2c, 3c in Appendix A). The model shows a stable pattern across the three corpora: While the vector groups representing different tags are spatially separated, the tag/no-tag instances are situated in very close proximity to each other and even partially overlap (*ja* and *nicht* in all corpora; *ne* in TW). The tag/no-tag distinction for *nicht* generally seems to be most definite, showing practically no overlap in OS and TW.[11]

This model also differentiates based on punctuation. In some cases, tags are divided into two distinct subgroups based on the end punctuation (*ne* and *ja* in CH). In most cases, though, the tags are ordered within their respective groups: Tags followed by a question mark and preceded by a comma are situated on one side of the vector group, whereas those ending with a period are placed on its other end. The latter is also where a (partial) overlap with the no-tags takes place, as no-tags are largely followed by a period.

[11]The clear tag/no-tag distinction for *nicht* is also made by the *bert-base-german-dbmdz-cased* model.

5.4. Summary

We find that *bert-base-german-dbmdz-cased* looks most promising for exploring tag interchangeability in our data. This model differentiates between the tags quite clearly, but also places several tag subgroups close to each other (contrary to *gbert-large*), which might indicate potential cases of similarity. Therefore, we perform a clustering analysis of its output (Section 6) and use this model for the tag word prediction task (Section 7).

6. Clustering

In this section, we focus only on the tag part of the data and apply the K-Means clustering algorithm to the BERT-generated tag vectors.[12] As discussed in the previous section, BERT groups tags by their form (and punctuation). By means of clustering, we explore whether there are any common features across these tag groups. Our assumption is that distinct tags that occur in similar contexts will have similar linguistic properties encoded in their vector representations and will hence be clustered together.

6.1. Cluster analysis

We experiment with different numbers of clusters (k) starting with 4 (i.e., the number of tags in the dataset) and increasing it in single steps up to 10. As discussed in Section 1, tags are interchangeable only in certain contexts, which is why we are interested in impure clusters, i.e., the ones where different tag groups are partially clustered together.

The general tendency we observe is that with higher k's, each tag form is allocated to a distinct cluster or even divided into multiple clusters. Hence, we determine the highest k (below 10) with which any different tags are still clustered together, and examine the resulting impure clusters in more detail. Following this strategy, we select k=9 for CallHome, k=7 for Twitter, and k=4 for OpenSubtitles. Table 3 shows the impure clusters. An overview of all clusters generated with the respective k's can be found in Figure 4 in Appendix A.

Table 3
Impure K-Means clusters per corpus. k denotes the overall number of clusters, { } mark cluster boundaries, subscript numbers indicate cluster IDs in plots.

Corpus	k	Impure clusters
CallHome	9	{part *nicht*, part *oder*}$_3$, {part *ne*, part *nicht*, 1 *oder*}$_7$
OpenSubtitles	4	{*ne*, 2 *nicht*, *oder*}$_4$
Twitter	7	{part *ja*, part *nicht*}$_2$

[12]We used the *scikit-learn* implementation: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html. The clusters are built on the original BERT vectors; the PCA-reduced vectors are used only for visualization purposes.

6.1.1. CallHome

Two impure clusters were generated with k=9 (Figure 4a in Appendix A). The cluster {part *nicht*, part *oder*}$_3$ contains the instances of these tags that are followed by a question mark and preceded by a comma. This makes up a part of the *oder*-subgroup and the complete *nicht*-subgroup with a question mark. A closer look at TQs in this corpus reveals that the ones with a question mark express requests for information or an opinion from the addressee (cf. questions and statement-question blends [12]).

The cluster {part *ne*, part *nicht*, 1 *oder*}$_7$ contains tags without the preceding comma and followed by a period (including occasional cases of alternative punctuation). This corresponds to a part of each respective tag's subgroup. TQs ending with a period in this corpus are those where the speaker has epistemic authority and provides information or an opinion.

We conclude that the clustering method supports the punctuation-based distinction among TQs, e.g., by utilizing the tag-preceding punctuation as a clustering criterion. The observed correlation between the end punctuation and certain TQ types can be attributed to the fact that CH contains transcribed data, where, evidently, question marks and periods represent the rising and falling intonation, respectively. This, in turn, corresponds (at least roughly) to the addressee vs. speaker epistemic authority. This correlation should be taken with a grain of salt, though, as it is not necessarily the case with other corpora, e.g., Twitter users do not follow punctuation rules strictly.

6.1.2. OpenSubtitles

One impure cluster – {*ne*, 2 *nicht*, *oder*}$_4$ – was generated with k=4 (Figure 4b in Appendix A). Any higher k merely led to multiple clusters for *ja* and *nicht*. This is not surprising, as these tags are represented by a notably larger number of instances than *ne* and *oder* in the corpus. This cluster comprises the total number of *ne* and *oder* in OS and covers a mix of different TQ types.

There is almost no variation in punctuation in this corpus: TQs without the tag-preceding comma and/or ending with a period make up less than 2% per tag. Due to this fractional amount, these cases are not decisive for the automatic analysis.

The homogeneous use of punctuation in this corpus might be explained by the fact that subtitles are supposed to conform with standard grammar (in our case, a tag separated from the anchor clause by a comma and followed by a question mark).

For this data, K-Means prioritizes the division of large tag groups into multiple clusters over the clustering of different tags together. We find no obvious differences between the instances of *ja* in the two clusters generated with k=4, e.g., they both contain directive TQs.

6.1.3. Twitter

One impure cluster was generated with k=7 (Figure 4c in Appendix A). This cluster – {part *ja*, part *nicht*}$_2$ – comprises the instances of the respective tags that have no preceding comma and are followed by a question mark. In Twitter, the question mark is the predominant end punctuation, and only few TQs end with a period (less than 1% with *nicht* and *oder*, 3% with *ja*,

and 15% with *ne*). Thus, tags are clustered based on the presence or absence of the preceding comma, rather than the end punctuation.

In general, K-Means merely assigns distinct clusters to the tag subgroups already formed by the BERT model. The clustering together of *nicht* with *ja* is not straightforward, especially since *oder* is situated closer to the former.

6.2. Mapping of linguistic properties to clusters

We assess how well the linguistic properties of TQs determined in the previous work map onto the K-means clusters. We use the annotations of the anchor clause type, anchor speech act, and target of confirmation from [4] available for a portion of the dataset used in this study: 940 TQs in CallHome and 641 TQs in Twitter.

To test the distribution of these properties across our clusters, we apply the cluster evaluation metric V-measure [18], which constitutes the harmonic mean between homogeneity (whether all TQs in a cluster belong to the same category, e.g., anchor clause type) and completeness (whether all TQs with the same properties are put into one cluster).[13] We find that the target of confirmation has the highest match with the clusters in both corpora: its V-measure scores range between 0.13-0.16 (CH and TW), depending on the number of clusters (between 4 and 10). The anchor clause type and speech act are both associated with lower scores: 0.05-0.09 (CH) and 0.11-0.16 (TW).

These results confirm previous observations that the tags' felicity conditions only partially depend on the anchor clause type and speech act. They also support previous findings that certain tags, such as *oder*, are infelicitous with requests to acknowledge the provided information, while other tags, such as *ne*, are typical for this target of confirmation [8].

7. Tag word prediction

In this section, we describe the BERT Masked Language Modeling task for the tag word prediction with the model selected in Section 5. We test the impact of fine-tuning on the model's performance and examine its predictions with regard to the tags' interchangeability potential. We implement the training task using PyTorch [16] and the HuggingFace Transformers library [21].

7.1. Experimental setup

For this task, we use the complete dataset (tags and no-tags) and fine-tune the BERT model to predict the tag word form (e.g., *ne* vs. *ja*) and class (tag vs. no-tag). We represent the no-tags with the special tokens [NTJA], [NTNE], and [NTNICHT] to differentiate them from the respective tags in the model's predictions.[14] The special tokens and tags are then replaced with the [MASK] token. We run the training for 10 epochs with standard parameters. The performance of the fine-tuned model is compared with that of the original pretrained model (baseline).

[13] We used the *scikit-learn* implementation: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.v_measure_score.html.

[14] The tag *oder* has no counterpart [NTODER].

The dataset is randomly split into the training and test sets (80% and 20% from each corpus, respectively). The training set is further randomly split into 80% training and 20% evaluation. We apply this configuration to (a) the whole dataset and (b) the dataset without OpenSubtitles in the training data. With this, we test how much the model relies on the OS data, which was part of its original pretraining.

Furthermore, we train the model separately on each corpus and test on the rest of the dataset. Our corpora differ in terms of style and conformity to standards: spoken telephone conversations (CH), transcribed spoken language (OS), and computer-mediated communication that can be placed somewhere between written and spoken (TW) [cf. 4]. With this, we test the suitability of different types of data for training a generalized model for tag prediction.

7.2. Evaluation

For each sentence, we consider the top three predictions and calculate two types of scores to evaluate the model's performance:

- SCORE_EQUAL – the model predicts the correct class (tag/no-tag) and the correct form (e.g., *ne*-tag for *ne*-tag)
- SCORE_CLOSE – the model predicts the correct class, but the form can be incorrect (e.g., *ja*-no-tag for *nicht*-no-tag or *ja*-tag for *nicht*-tag); this score includes SCORE_EQUAL

We sum up the probabilities that match these criteria within the top three predictions to obtain a single score. The calculation is demonstrated below for a TQ from the Twitter corpus in (4):

(4) *Eh Digga, das war voll fett krass alter **oder**?*
'Eh dude, that was absolutely totally cool man right?'

The top three predictions and their probabilities for this TQ are *oder* (0.933), *ne* (0.03), and [NTNICHT] (0.026). Thus, SCORE_EQUAL amounts to 0.933 + 0 + 0 = 0.933 (93%) and SCORE_CLOSE to 0.933 + 0.03 + 0 = 0.963 (96%). Additionally, we report precision, recall, and F1 scores based on the model's top prediction.

7.3. Results

The SCORE_EQUAL and SCORE_CLOSE results are given in Table 4. Independently of whether OS is present in the training data, we observe a considerable improvement over the baseline (both scores). The tag/no-tag distinction (SCORE_CLOSE) reaches almost a 100% probability in most cases.

With OS in the training data, the lowest SCORE_EQUAL values are obtained for *ne*<*oder*<*nicht* (increasing in this order). This reflects the number of the respective tags in the training part of the dataset, with less frequent tags receiving poorer scores. The baseline scores are distributed differently, suggesting that *oder* and *ja* were the most frequent tags in the model's original training data. However, the correctness probability of the baseline model does not go beyond

Table 4
Experiment results: training on all corpora (upper part) and without OS (lower part). BL stands for baseline, FT for fine-tuned.

			EQUAL (%)		CLOSE (%)	
	# test	Tag word	BL	FT	BL	FT
CH+TW+OS	2,457	*ja*	17	84	36	99
	351	*ne*	0	31	22	86
	1,068	*nicht*	8	59	46	99
	524	*oder*	31	43	37	94
	4,400	all tags	15	69	37	97
	10,908	tags + no-tags	6	86	15	98
CH+TW	2,481	*ja*	17	55	36	99
	331	*ne*	0	43	22	89
	1,075	*nicht*	7	18	47	99
	531	*oder*	30	67	37	95
	4,418	all tags	15	47	37	98
	10,909	tags + no-tags	6	76	15	98

50% (both scores). Given that we introduced the no-tag special tokens for this task, the baseline scores are especially low in the test containing all items (tags and no-tags).

Without OS in the training data, SCORE_EQUAL drops drastically for *nicht* and *ja*. We attribute this to the fact that the majority of TQs with these tags come from this corpus, thus limiting the model's exposure to this type of data during training. The importance of large datasets for predictions with BERT was emphasized in previous studies [e.g., 13, 1].

Precision, recall, and F1 scores show a (notable) improvement of the fine-tuned model over the baseline for each tag (Tables 5 and 6). When trained on all corpora, the fine-tuned model shows lower recall for *oder* compared to the baseline. The latter provides reasonable results primarily for *ja*. Its predictions for *ne* and *nicht* tend towards zero.

Table 5
Experiment results for training on all corpora: precision (P), recall (R), and F1 scores.

		P (%)		R (%)		F1 (%)	
# test	Tag word	BL	FT	BL	FT	BL	FT
316	NTJA	0	73	0	61	0	67
11	NTNE	0	67	0	18	0	29
6,181	NTNICHT	0	97	0	99	0	98
2,457	*ja*	77	81	33	84	46	83
351	*ne*	0	46	0	32	0	38
1,068	*nicht*	1	60	3	60	1	60
524	*oder*	15	51	59	44	24	47

The experiments with training on one corpus and testing on the rest of the dataset resulted in a lower performance compared to the training on the data from all corpora. This can be

Table 6
Experiment results for training without OS: precision (P), recall (R), and F1 scores.

		P (%)		R (%)		F1 (%)	
# test	Tag word	BL	FT	BL	FT	BL	FT
329	NTJA	0	70	0	54	0	61
12	NTNE	0	33	0	8	0	13
6,150	NTNICHT	0	97	0	99	0	98
2,481	*ja*	76	89	33	56	46	68
331	*ne*	0	20	0	43	0	27
1,075	*nicht*	1	56	2	18	1	27
531	*oder*	16	20	60	67	25	31

explained by the limited amount of the training data (CH in particular turned out to be least suitable for training). Another reason is that our data, especially OS and TW, is imbalanced and certain tags are heavily underrepresented. As with the tests described above, the results here directly depend on the amount of the training data: The tag words represented by larger numbers of instances received higher scores.

In addition to these tests, we examine the top three predictions in the results of the training on all corpora (see Section 7.1) regarding the frequency with which different tags were suggested by BERT for each original tag variant.[15] We hope to find indications of the tags' interchangeability by examining which tags might constitute the best substitutes for each other. For TQs with *ne*, BERT predicted *ne*, *ja*, and *nicht* with almost equal frequency (in 21-23% of the cases for each). For TQs with *ja*, *nicht*, or *oder*, the original tag was predicted in the majority of the cases (27-32%, depending on the tag). The next-best alternatives were as follows: *nicht* (29%) for *ja*, *ja* (25%) for *nicht*, and both *ja* and *nicht* (21% each) for *oder*. These results suggest that *oder* and *ne* are generally poor substitutes for each other, which confirms previous corpus-based results [4]. The indication that *ne* could be replaced by *nicht* or *ja* is consistent with the experimental evidence in [5], which shows that these tags have common characteristics. For example, they are less felicitous in TQs expressing speaker assumptions based on the addressee's behavior.

8. Discussion and conclusion

This study explored whether the differences among the four common German tags *ja*, *ne*, *nicht*, and *oder*, such as those established in previous corpus-based and experimental work, can be interpreted and predicted automatically. Our analysis of the existing German BERT models showed that they strongly depend on structural features, such as the tag-surrounding punctuation. For example, tags and no-tags were oftentimes regarded as more similar to one another than to other instances of the respective classes due to matching punctuation, while syntactic

[15]We look at frequencies instead of probabilities, as in our data the latter are typically considerably lower for the second and third top predictions compared to the first one. This might be different with a larger dataset, though.

and semantic properties of TQs were not recognizably detected.

We examined the tag vectors generated by one of these models in more detail. The mapping of linguistic properties of TQs to the automatically formed clusters of the tag vectors confirmed previous observations that the target of confirmation is a more informative feature for tags' differentiation than, for instance, the syntactic properties of the TQ anchor.

Furthermore, we fine-tuned the selected model on the tag word prediction task. The tag word class (tag/no-tag) was predicted with near 100% probability in most cases. The prediction of the tag word form proved to be more challenging, though. Especially the experiments with training on single corpora highlighted the importance of the dataset size: The predicted tag word probabilities directly correlated with the number of instances they were represented by in the training set. Overall, the results showed that with standard parameters and given a large enough training dataset (14,045 tags and 20,860 no-tags, in our case) the fine-tuned model works well for this task. However, hyper-parameter optimization and class weighting are worth exploring in the future.

The difficulties with the automatic distinction between the tag forms are not overly surprising, after all. Cases where different TQ types share syntactic and semantic properties of the anchor provide limited information for BERT to rely on in order to, for example, rule out the use of certain tags, such as *oder* in informing TQs. The absence of additional contextual information hinders the judgments about the tags' felicity in such cases. Nonetheless, certain TQs contain sufficient information to predict the tag even without context, e.g., *ja* in imperative directives. Since they differ both semantically and syntactically from TQs with declarative anchors, we would expect BERT to pick up on their specific properties. However, possibly because of their underrepresentation in our dataset, these TQs were not identified. Augmentation of the dataset with certain (synthetically generated) TQ types would facilitate further testing of BERT's capacity to detect their features.

We conclude that BERT provides indications of TQ features that are useful for tag differentiation. It also seems to correctly recognize which tags constitute appropriate substitutes for each other, although this needs further testing on a larger dataset. In future work, it could be worth including the right-side context of the tags (not present in our data) to fully exploit the power of BERT to use bidirectional context.

Acknowledgments

We thank Tatjana Scheffler and Manfred Stede for discussions and valuable suggestions. We are grateful to the anonymous reviewers for their helpful comments.

This research was funded by the PhD completion scholarship from the Graduate Fund of the State of Brandenburg awarded by the University of Potsdam, and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), CRC 1567, Project ID 470106373.

References

[1] F. Bianchi, B. Yu, and J. Tagliabue. "BERT Goes Shopping: Comparing Distributional Models for Product Representations". In: *Proceedings of the 4th Workshop on e-Commerce*

and NLP. Online: Association for Computational Linguistics, 2021, pp. 1–12. DOI: 10.18653/v1/2021.ecnlp-1.1.

[2] B. Chan, S. Schweter, and T. Möller. "German's Next Language Model". In: *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, 2020, pp. 6788–6796. DOI: 10.18653/v1/2020.coling-main.598.

[3] Y. Clausen. "You shall know a tag by the context it occurs in: An analysis of German tag questions and their responses in spontaneous conversations". In: *ConSOLE XXIX: Proceedings of the 29th Conference of the Student Organization of Linguistics in Europe*. Ed. by A. Holtz, I. Kovač, R. Puggaard-Rode, and J. Wall. Leiden: Leiden University Centre for Linguistics, 2021, pp. 116–140. URL: https://www.universiteitleiden.nl/binaries/content/assets/geesteswetenschappen/lucl/sole/consolexxix.pdf.

[4] Y. Clausen and T. Scheffler. "A corpus-based analysis of meaning variations in German tag questions: Evidence from spoken and written conversational corpora". In: *Corpus Linguistics and Linguistic Theory* 18.1 (2022), pp. 1–31. DOI: 10.1515/cllt-2019-0060.

[5] Y. Clausen and T. Scheffler. "Commitments in German Tag Questions: An Experimental Study". In: *Proceedings of the 24th Workshop on the Semantics and Pragmatics of Dialogue - Full Papers*. Virtually at Brandeis, Waltham, New Jersey: Semdial, 2020. URL: http://semdial.org/anthology/Z20-Clausen%5C%5Fsemdial%5C%5F0014.pdf.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by J. Burstein, C. Doran, and T. Solorio. Minneapolis, MN, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/n19-1423.

[7] A. Ettinger. "What BERT Is Not: Lessons from a New Suite of Psycholinguistic Diagnostics for Language Models". In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 34–48. DOI: 10.1162/tacl_a_00298.

[8] J. Hagemann. "Tag questions als Evidenzmarker. Formulierungsdynamik, sequentielle Struktur und Funktionen redezuginterner tags". In: *Gesprächsforschung – Online-Zeitschrift zur verbalen Interaktion* 10 (2009), pp. 145–176.

[9] J. Heim. "Turn-peripheral management of Common Ground: A study of Swabian *gell*". In: *Journal of Pragmatics* 141 (2019), pp. 130–146. DOI: 10.1016/j.pragma.2018.12.007.

[10] K. Karins, R. MacIntyre, M. Brandmair, S. Lauscher, and C. McLemore. *CALLHOME German Transcripts LDC97T15*. Web Download. Philadelphia: Linguistic Data Consortium. 1997. URL: https://catalog.ldc.upenn.edu/LDC97T15.

[11] T. Kato, R. Miyata, and S. Sato. "BERT-Based Simplification of Japanese Sentence-Ending Predicates in Descriptive Text". In: *Proceedings of the 13th International Conference on Natural Language Generation*. Dublin, Ireland: Association for Computational Linguistics, 2020, pp. 242–251. URL: https://aclanthology.org/2020.inlg-1.31.

[12] D. Kimps, K. Davidse, and B. Cornillie. "A speech function analysis of tag questions in British English spontaneous dialogue". In: *Journal of Pragmatics* 66 (2014), pp. 64–85. DOI: doi.org/10.1016/j.pragma.2014.02.013.

[13] W. Li, A. Grissom II, and J. Boyd-Graber. "An Attentive Recurrent Model for Incremental Prediction of Sentence-final Verbs". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, 2020, pp. 126–136. DOI: 10.18653/v1/2020.findings-emnlp.12.

[14] P. Lison and J. Tiedemann. "OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. Portorož, Slovenia: European Language Resources Association (ELRA), 2016, pp. 923–929. URL: https://aclanthology.org/L16-1147.

[15] S. Mandokoro, N. Oka, A. Matsushima, C. Fukada, Y. Yoshimura, K. Kawahara, and K. Tanaka. "Construction and Evaluation of a Self-Attention Model for Semantic Understanding of Sentence-Final Particles". In: *arXiv preprint* (2022). DOI: 10.48550/arXiv.2210.00282.

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'AlchéBuc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[17] A. Rogers, O. Kovaleva, and A. Rumshisky. "A Primer in BERTology: What We Know About How BERT Works". In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 842–866. URL: https://aclanthology.org/2020.tacl-1.54.

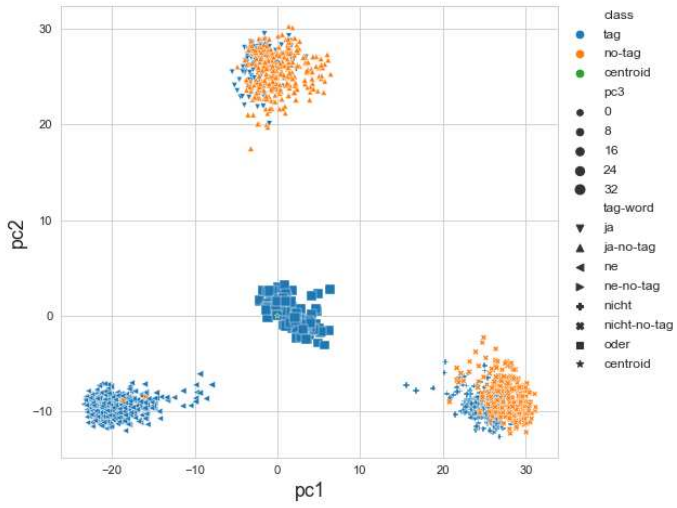
[18] A. Rosenberg and J. Hirschberg. "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure". In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic: Association for Computational Linguistics, 2007, pp. 410–420. URL: https://aclanthology.org/D07-1043.

[19] T. Scheffler. "A German Twitter Snapshot". In: *Proceedings of the 19th International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), 2014, pp. 2284–2289. URL: http://www.lrec-conf.org/proceedings/lrec2014/pdf/1146%5C%5FPaper.pdf.

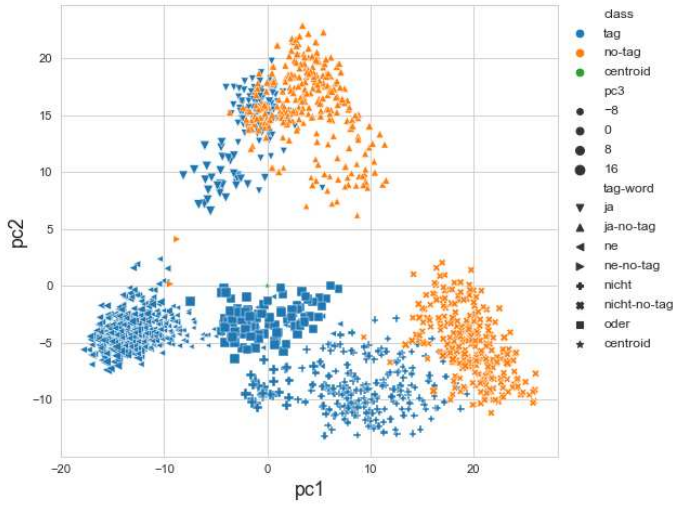
[20] M. Wiltschko, D. Denis, and A. D'Arcy. "Deconstructing variation in pragmatic function: A transdisciplinary case study". In: *Language in Society* 47.4 (2018), pp. 569–599. DOI: 10.1017/s004740451800057x.

[21] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

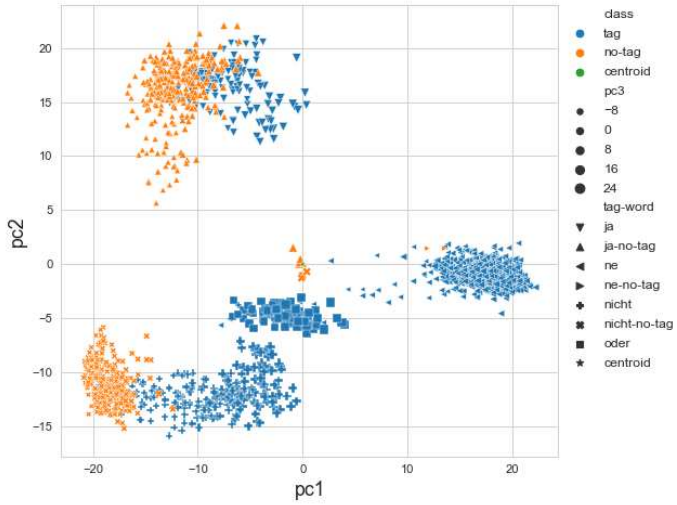
A. Visualization of BERT Vectors



(a) *bert-base-german-cased*

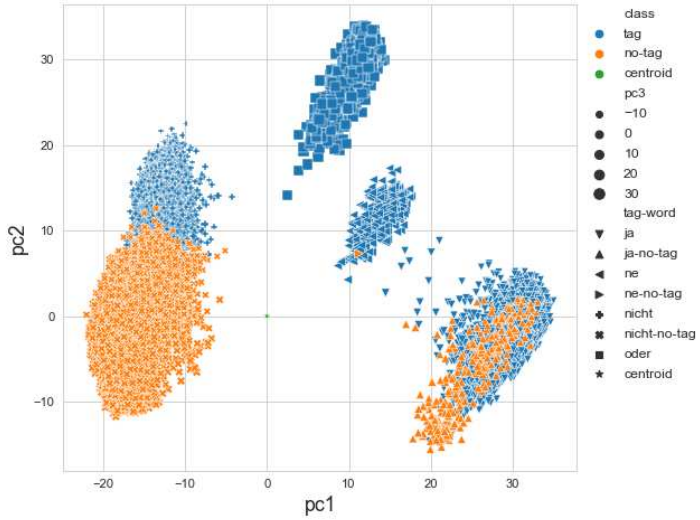


(b) *bert-base-german-dbmdz-cased*

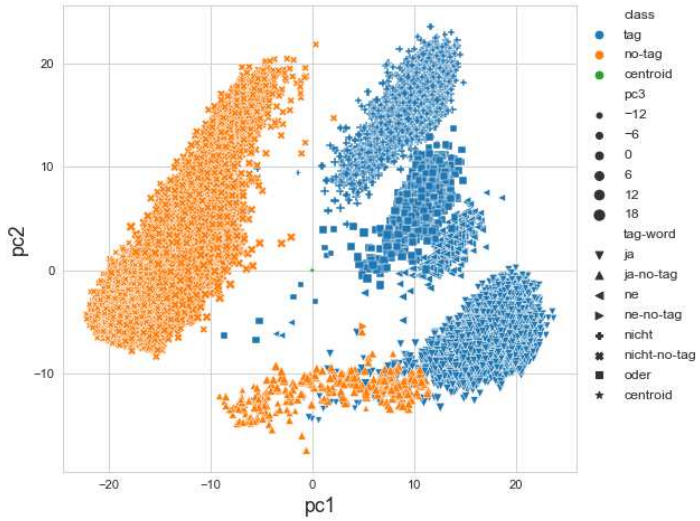


(c) *gbert-large*

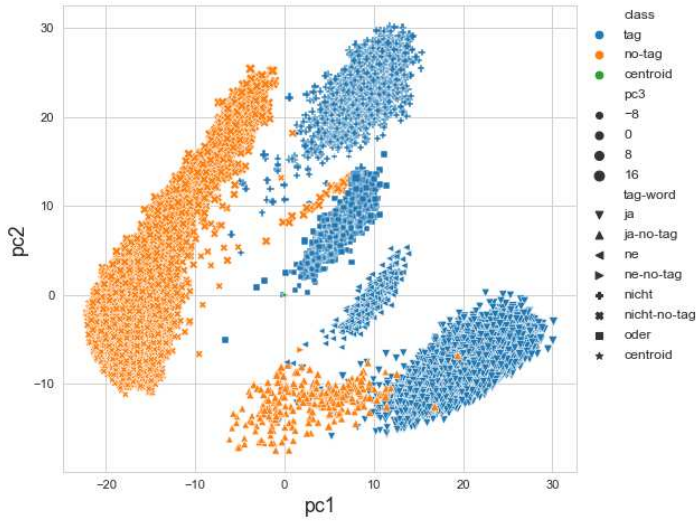
Figure 1: BERT vectors for the tag words *ja*, *ne*, *nicht*, and *oder* in the CallHome corpus. In all plots, *pc3* represents the z-axis.



(a) *bert-base-german-cased*

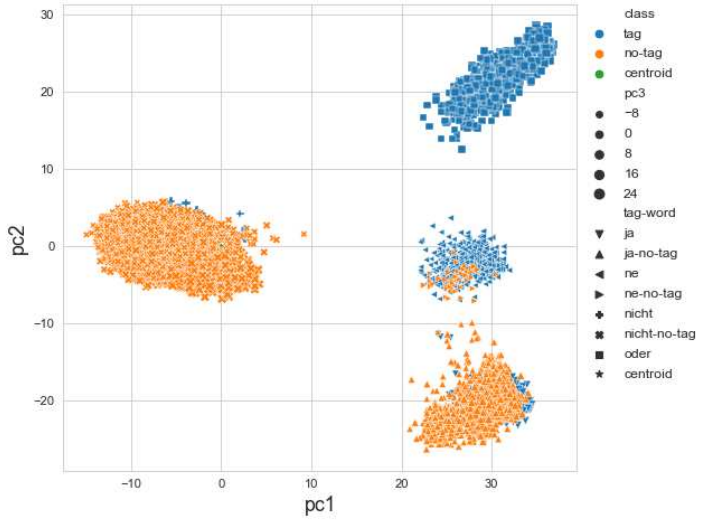


(b) *bert-base-german-dbmdz-cased*

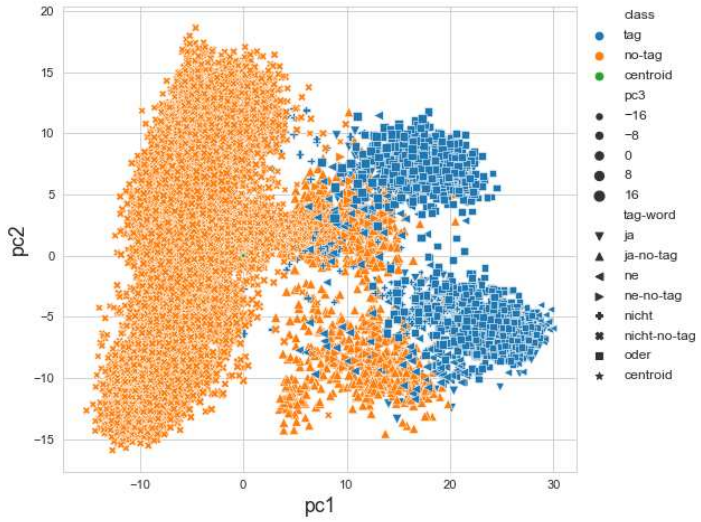


(c) *gbert-large*

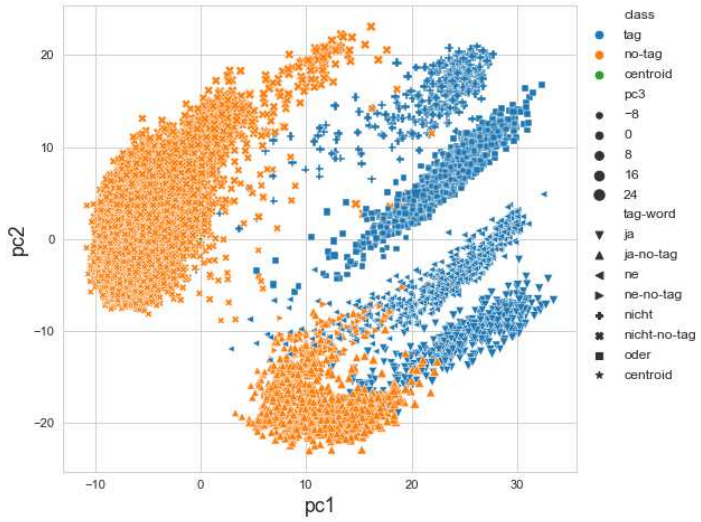
Figure 2: BERT vectors for the tag words *ja*, *ne*, *nicht*, and *oder* in the OpenSubtitles corpus.



(a) *bert-base-german-cased*

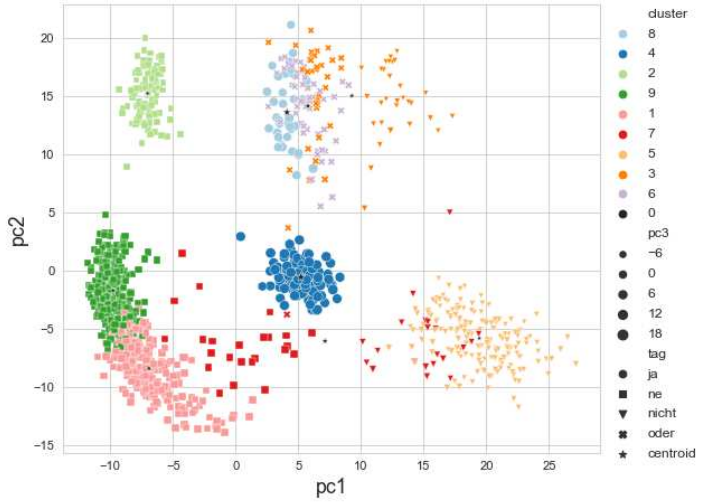


(b) *bert-base-german-dbmdz-cased*

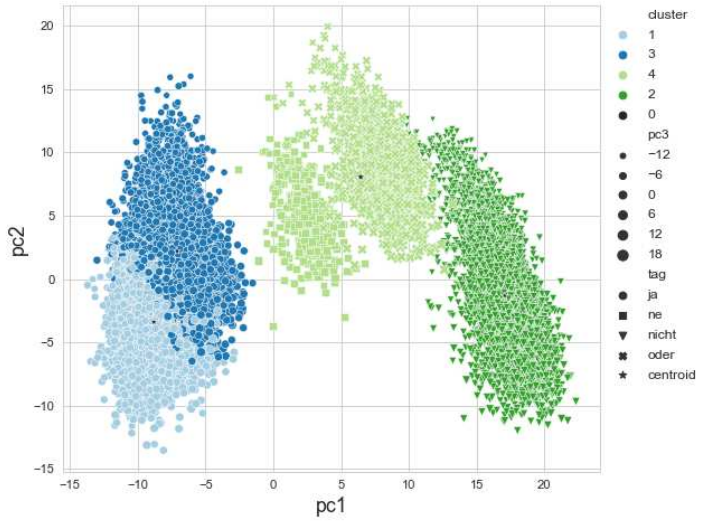


(c) *gbert-large*

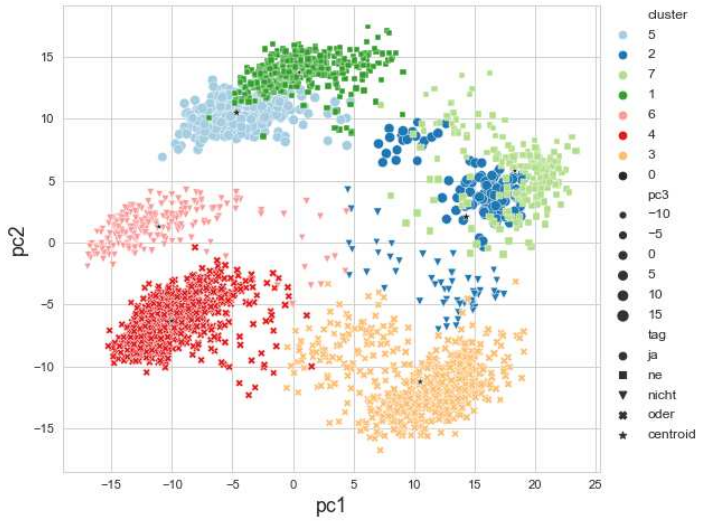
Figure 3: BERT vectors for the tag words *ja*, *ne*, *nicht*, and *oder* in the Twitter corpus.



(a) CallHome, k=9



(b) OpenSubtitles, k=4



(c) Twitter, k=7

Figure 4: K-Means clusters for the *bert-base-german-dbmdz-cased* tag vectors. Cluster 0 represents the centroid.